

日本国特許庁
JAPAN PATENT OFFICE

H. Koyanagi
7/22/03
Q 76637 1071

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出願年月日
Date of Application:

2002年 7月25日

出願番号
Application Number:

特願2002-216607

[ST.10/C]:

[JP 2002-216607]

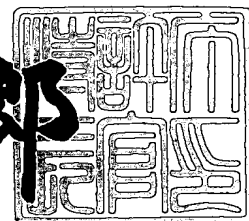
出願人
Applicant(s):

日本電気株式会社

2003年 5月 6日

特許庁長官
Commissioner,
Japan Patent Office

太田信一郎



出証番号 出証特2003-3032931

【書類名】 特許願

【整理番号】 66206604

【提出日】 平成14年 7月25日

【あて先】 特許庁長官 殿

【国際特許分類】 G06F 9/38

【発明者】

【住所又は居所】 東京都港区芝五丁目7番1号 日本電気株式会社内

【氏名】 小柳 尚夫

【特許出願人】

【識別番号】 000004237

【氏名又は名称】 日本電気株式会社

【代理人】

【識別番号】 100088328

【弁理士】

【氏名又は名称】 金田 暢之

【電話番号】 03-3585-1882

【選任した代理人】

【識別番号】 100106297

【弁理士】

【氏名又は名称】 伊藤 克博

【選任した代理人】

【識別番号】 100106138

【弁理士】

【氏名又は名称】 石橋 政幸

【手数料の表示】

【予納台帳番号】 089681

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9710078

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 情報処理装置及びそのメモリ制御方法

【特許請求の範囲】

【請求項 1】 ベクトル方式の情報処理装置であって、
複数の非同期動作ユニットから構成される CPU と、
データを格納するメインメモリ部と、

ベクトル・スキヤッタ命令で指定された各要素の前記メインメモリ部への格納先アドレスを保持する V S C アドレスバッファを備え、ベクトル・スキヤッタ命令で前記非同期動作ユニットからそれぞれ発行される前記メインメモリ部に対する要素の書き込みリクエストのうち、前記格納先アドレスが一致し、かつ要素番号順に処理されていない場合に、要素番号が若い要素の書き込みリクエストに対応して生成される前記メインメモリ部への書き込み許可信号の出力を抑止する、前記メインメモリ部に対するデータの書き込み処理を制御するメインメモリ制御部と、
を有する情報処理装置。

【請求項 2】 前記メインメモリ制御部は、
前記複数の非同期動作ユニットに対応する複数の V S C アドレスバッファを備えた請求項 1 記載の情報処理装置。

【請求項 3】 前記非同期動作ユニットは、
1 つのベクトル・スキヤッタ命令で発行される複数の前記書き込みリクエストに対して同一の識別子を付与し、
前記メインメモリ制御部は、
先行する書き込みリクエストの識別子と後続の書き込みリクエストの識別子が一致しない場合に、前記 V S C アドレスバッファの内容をクリアする請求項 2 記載の情報処理装置。

【請求項 4】 前記メインメモリ制御部は、
前記 V S C アドレスバッファで前記複数の非同期動作ユニットから送信される前記格納先アドレスをそれぞれ保持し、前記 V S C アドレスバッファがオーバーフローを起こした場合に、該オーバーフローさせたベクトル・スキヤッタ命令を

発行した非同期動作ユニットへ前記要素の再送を要求する V S C アドレスバッファ制御部を備え、

前記非同期動作ユニットは、

自ユニットが発行した前記ベクトル・スキヤッタ命令で指定された各要素をそれぞれ保持するリトライバッファを有し、前記メインメモリ制御部から前記要素の再送が要求された場合に、前記リトライバッファに保持した要素を前記メインメモリ制御部へ再送する請求項 1 記載の情報処理装置。

【請求項 5】 前記非同期動作ユニットは、

各非同期動作ユニットで再送を開始する要素のうち、最も若い要素番号に基づいて再送を開始する要素の要素番号を修正する請求項 4 記載の情報処理装置。

【請求項 6】 前記メインメモリ制御部は、

前記 V S C アドレスバッファのオーバーフローと前記非同期動作ユニットからの要素の再送を繰り返し実行するデッドロック状態を検出した場合に、前記非同期動作ユニットからの要素の再送タイミングをずらすための遅延値を前記非同期動作ユニットへ送出し、

前記非同期動作ユニットは、

前記要素の再送タイミングを前記メインメモリ制御部から受信した該遅延値だけ遅らせる請求項 4 または 5 記載の情報処理装置。

【請求項 7】 前記 V S C アドレスバッファで保持する前記格納先アドレスの数は、前記非同期動作ユニットが一度に処理する要素数 + 1 以上である請求項 4 乃至 6 のいずれか 1 項記載の情報処理装置。

【請求項 8】 複数の非同期動作ユニットから構成される C P U と、

データを格納するメインメモリ部と、

前記メインメモリ部に対するデータの書き込み処理を制御するメインメモリ制御部とを有するベクトル方式の情報処理装置のメモリ制御方法であって、

予め、前記メインメモリ制御部に、ベクトル・スキヤッタ命令で指定された各要素の前記メインメモリ部への格納先アドレスを保持するための V S C アドレスバッファを備えておき、

前記ベクトル・スキヤッタ命令で前記非同期動作ユニットからそれぞれ発行さ

れる前記メインメモリ部に対する要素の書き込みリクエストのうち、前記格納先アドレスが一致し、かつ要素番号順に処理されていない場合に、要素番号が若い書き込みリクエストに対応して生成される前記メインメモリ部への書き込み許可信号の出力を抑止するメモリ制御方法。

【請求項 9】 1つのベクトル・スキヤッタ命令で指定された複数要素の格納先アドレスのみ前記VSCアドレスバッファで保持する請求項8記載のメモリ制御方法。

【請求項 10】 1つのベクトル・スキヤッタ命令で発行される前記書き込みリクエストに対してそれぞれ同一の識別子を付与し、

先行する前記書き込みリクエストの識別子と後続の前記書き込みリクエストの識別子が一致しない場合に、前記VSCアドレスバッファの内容をクリアする請求項8または9記載のメモリ制御方法。

【請求項 11】 前記複数の非同期動作ユニットから送信される前記格納先アドレスをVSCアドレスバッファでそれぞれ保持し、

前記VSCアドレスバッファがオーバーフローを起こした場合に、該オーバーフローさせたベクトル・スキヤッタ命令を発行した非同期動作ユニットへ前記要素の再送を要求し、

前記非同期動作ユニットで、発行した前記ベクトル・スキヤッタ命令で指定された各要素をそれぞれ保持しておき、前記要素の再送が要求された場合は、前記リトライバッファに保持した要素を前記メインメモリ制御部へ再送する請求項8記載のメモリ制御方法。

【請求項 12】 各非同期動作ユニットで再送を開始する要素のうち、最も若い要素番号に基づいて再送を開始する要素の要素番号を修正する請求項11記載のメモリ制御方法。

【請求項 13】 前記VSCアドレスバッファのオーバーフローと前記非同期動作ユニットからの要素の再送を繰り返し実行するデッドロック状態を検出した場合に、前記非同期動作ユニットからの要素の再送タイミングをずらすための遅延値を前記メインメモリ制御部から前記非同期動作ユニットへ送出し、

前記非同期動作ユニットは、前記要素の再送タイミングを前記メインメモリ制

御部から受信した該遅延値だけ遅らせる請求項 1 1 または 1 2 記載のメモリ制御方法。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、情報処理装置に関し、特にハードウェアで並列処理を実現するベクトル方式の情報処理装置に関する。

【0 0 0 2】

【従来の技術】

近年の情報処理装置は、その動作周波数の高速化に伴って伝送線路による信号遅延が大きくなり、複数の半導体集積回路装置（CPU／LSI等）を位相の一致したクロックで動作させることが非常に困難になってきている。

【0 0 0 3】

このような問題の一つの解決策として、非同期クロックで動作する複数のCPUの処理をソフトウェアで同期させる手法がある。例えば、バリア同期／通信レジスタと呼ばれるハードウェア機能を利用して、指定された複数のプロセスを異なるOS（Operating System）で動作するCPUヘディスパッチ(dispatch)する方法がある。この方法は、複数のプロセスが全く異なるタイミングで動作することを前提としているため、各CPUのクロックが非同期であってもハードウェア機能に起因する動作不良が発生しない。このような方法は、スカラ型並列コンピュータと呼ばれる製品等で実現されている。

【0 0 0 4】

上記のようなソフトウェアで複数のプロセスを同期させる手法では、CPUの動作速度やCPUとメモリ間のデータ転送速度の高速化等のように、ハードウェア性能を向上させる手法と比較して非常に安価に実現できるため、見かけ上の性能対価格比が向上する。

【0 0 0 5】

しかしながら、ソフトウェアで複数のプロセスを同期させる手法ではプログラムの並列化が非常に困難であるという問題がある。これは、プログラムで用いる

各命令の並列化限界が千差万別であることに起因する。仮に並列化できても、そのデバッグ作業は、並列化されていないプログラムよりも格段に困難になる。デバッグ作業は、情報処理装置の性能チューニング時に行うのが一般的であるが、並列処理技術に関する高度なスキルを必要とする。また、このような困難なデバッグ作業をハードウェアがバージョンアップする度に行う必要があり、プログラムの膨大な資産を活かし難いという問題もある。さらに、プログラムの並列化のための技術的な課題を克服しても、それを現場で運用する際に人的リソース不足等の問題も発生する。

【0006】

このような問題の解決策の1つとしてハードウェアによる並列処理があり、その具体例としてベクトル方式の情報処理装置がある。

【0007】

ベクトル方式とは、規則的に並んだ複数個の配列データを一度に処理する方式 (Single Instruction Multiple Data stream : SIMD型) であり、その複数個のデータを格納するレジスタはベクトルレジスタと呼ばれ、ベクトルレジスタに格納された全ての要素に対して、同一の演算、メモリアクセス、転送等を指示する命令はベクトル命令と呼ばれる。

【0008】

ベクトル命令は、例えば、

LVL VL<-128

VADD V7 <- V5+V4

のように記載される。この例は、VL (ベクトル・レンジ・レジスタ) に処理対象の要素 (要素数128) をLVL (Load VL) 命令を用いて格納し、その後、VADD (ベクトル加算) 命令を用いてベクトルレジスタV5、V4の各要素 (128個) を加算し、加算結果をベクトルレジスタV7に格納する処理である。

【0009】

ベクトル方式では、ソフトウェアによるプロセス間の同期が不要であるため、単一のCPUと同様の考え方でソフトウェアを作成することができる。また、並

列化手法として実績があり、ベクトル化のためのコンパイラも既に存在する。

【0010】

なお、このようなベクトル方式においても、性能を向上させるためには、それに見合ったCPUとメモリ間のバンド幅（データ転送速度）を確保する必要がある。また、CPUを、ベクトル命令を実行する複数のベクトルユニットで構成し、各ユニットを並列に動作させれば、演算処理等をより高速に実行することも可能である。

【0011】

【発明が解決しようとする課題】

上述のような複数のベクトルユニットで構成されたCPUを有するベクトル方式の情報処理装置では、VSC（ベクトル・スキッタ）命令の実行時に以下に記載する問題が発生する。

【0012】

VSC命令は、ベクトル方式の情報処理装置において非常に重要な命令である。その命令仕様について図8を用いて説明する。

【0013】

図8に示すように、VSC命令は、Yフィールドで指定されたベクトルレジスタVyの要素をアドレスとして用い、それに対応するZフィールドで指定されたベクトルレジスタVzの要素をメモリに格納する命令である。なお、図8のOPCフィールドはVSC命令であることを示すオペコードであり、Xフィールドは使用されない無効領域である。

【0014】

VSC命令に対する処理では、メモリへの各要素の書き込み処理が要素番号順に実行されなければならない。特に、同じアドレスに対して複数の要素を格納する場合は要素番号が大きい要素の書き込み処理を優先させなければならない。例えば、要素nと要素n+1とを同じメモリアドレスに格納する場合、要素n+1の書き込みを優先させて要素nを無効化する必要がある。このような制限は、従来のように1つのユニットまたは同期して動作する複数のユニットで処理する場合は、1つのポートから要素番号順に書き込みリクエストが発行されるため、特

に考慮する必要は無かった。

【0015】

しかしながら、CPUが複数の非同期動作ユニットで構成されるベクトル方式の情報処理装置では、各ユニットから発行される要素の書き込みリクエスト（以下、要素リクエストと称す）による処理の順番が保証されないため、メモリに対する要素の書き込み順序が逆転することがある。

【0016】

例えば、図9に示すように、CPUが、非同期動作ユニットであるマスターユニット（Master）とスレーブユニット（Slave）で構成され、隣り合った要素番号（要素nと要素n+1）の要素リクエストがそれらに分配されて発行される場合、要素nと要素n+1を同じメモリアドレスに格納する要素リクエストであると、メモリへの書き込み動作を制御するメモリ制御部で要素nの処理よりも要素n+1の処理を先に実行する可能性がある。そのため、要素n+1が要素nで上書きされてしまう。

【0017】

このような問題の解決策として、複数の非同期動作ユニットから発行される各要素リクエストを同期させる手法が考えられる。しかしながら、そのような方法では、同期処理のためのオーバーヘッドが大きくなり、要素リクエストの発行間隔が広がるため、マスターユニットとスレーブユニットとを並列動作させることによる処理の高速化の効果をうち消してしまう。

【0018】

本発明は上記したような従来の技術が有する問題点を解決するためになされたものであり、CPUが複数の非同期動作ユニットで構成される場合に、各ユニット間の処理順序の逆転による影響を排除することが可能な情報処理装置及び方法を提供することを目的とする。

【0019】

【課題を解決するための手段】

上記目的を達成するため本発明の情報処理装置は、ベクトル方式の情報処理装置であって、

複数の非同期動作ユニットから構成されるCPUと、

データを格納するメインメモリ部と、

ベクトル・スキヤッタ命令で指定された各要素の前記メインメモリ部への格納先アドレスを保持するVSCアドレスバッファを備え、ベクトル・スキヤッタ命令で前記非同期動作ユニットからそれぞれ発行される前記メインメモリ部に対する要素の書き込みリクエストのうち、前記格納先アドレスが一致し、かつ要素番号順に処理されていない場合に、要素番号が若い要素の書き込みリクエストに対応して生成される前記メインメモリ部への書き込み許可信号の出力を抑止する、前記メインメモリ部に対するデータの書き込み処理を制御するメインメモリ制御部と、

を有する構成である。

【0020】

このとき、前記メインメモリ制御部は、

前記複数の非同期動作ユニットに対応する複数のVSCアドレスバッファを備えていてもよく、

前記非同期動作ユニットは、

1つのベクトル・スキヤッタ命令で発行される複数の前記書き込みリクエストに対して同一の識別子を付与し、

前記メインメモリ制御部は、

先行する書き込みリクエストの識別子と後続の書き込みリクエストの識別子が一致しない場合に、前記VSCアドレスバッファの内容をクリアしてもよい。

【0021】

また、前記メインメモリ制御部は、

前記VSCアドレスバッファで前記複数の非同期動作ユニットから送信される前記格納先アドレスをそれぞれ保持し、前記VSCアドレスバッファがオーバーフローを起こした場合に、該オーバーフローさせたベクトル・スキヤッタ命令を発行した非同期動作ユニットへ前記要素の再送を要求するVSCアドレスバッファ制御部を備え、

前記非同期動作ユニットは、

自ユニットが発行した前記ベクトル・スキヤッタ命令で指定された各要素をそれぞれ保持するリトライバッファを有し、前記メインメモリ制御部から前記要素の再送が要求された場合に、前記リトライバッファに保持した要素を前記メインメモリ制御部へ再送してもよい。

【 0 0 2 2 】

このとき、前記非同期動作ユニットは、

各非同期動作ユニットで再送を開始する要素のうち、最も若い要素番号に基づいて再送を開始する要素の要素番号を修正してもよく、

前記メインメモリ制御部は、

前記VSCアドレスバッファのオーバーフローと前記非同期動作ユニットからの要素の再送を繰り返し実行するデッドロック状態を検出した場合に、前記非同期動作ユニットからの要素の再送タイミングをずらすための遅延値を前記非同期動作ユニットへ送出し、

前記非同期動作ユニットは、

前記要素の再送タイミングを前記メインメモリ制御部から受信した該遅延値だけ遅らせてもよい。

【 0 0 2 3 】

なお、前記VSCアドレスバッファで保持する前記格納先アドレスの数は、前記非同期動作ユニットが一度に処理する要素数+1以上であることが望ましい。

【 0 0 2 4 】

一方、本発明のメモリ制御方法は、複数の非同期動作ユニットから構成されるCPUと、

データを格納するメインメモリ部と、

前記メインメモリ部に対するデータの書き込み処理を制御するメインメモリ制御部とを有するベクトル方式の情報処理装置のメモリ制御方法であって、

予め、前記メインメモリ制御部に、ベクトル・スキヤッタ命令で指定された各要素の前記メインメモリ部への格納先アドレスを保持するためのVSCアドレスバッファを備えておき、

前記ベクトル・スキヤッタ命令で前記非同期動作ユニットからそれぞれ発行さ

れる前記メインメモリ部に対する要素の書き込みリクエストのうち、前記格納先アドレスが一致し、かつ要素番号順に処理されていない場合に、要素番号が若い書き込みリクエストに対応して生成される前記メインメモリ部への書き込み許可信号の出力を抑止する方法である。

【 0 0 2 5 】

このとき、1つのベクトル・スキヤッタ命令で指定された複数要素の格納先アドレスのみ前記VSCアドレスバッファで保持してもよく、

1つのベクトル・スキヤッタ命令で発行される前記書き込みリクエストに対してそれぞれ同一の識別子を付与し、

先行する前記書き込みリクエストの識別子と後続の前記書き込みリクエストの識別子が一致しない場合に、前記VSCアドレスバッファの内容をクリアしてもよい。

【 0 0 2 6 】

また、前記複数の非同期動作ユニットから送信される前記格納先アドレスをVSCアドレスバッファでそれぞれ保持し、

前記VSCアドレスバッファがオーバーフローを起こした場合に、該オーバーフローさせたベクトル・スキヤッタ命令を発行した非同期動作ユニットへ前記要素の再送を要求し、

前記非同期動作ユニットで、発行した前記ベクトル・スキヤッタ命令で指定された各要素をそれぞれ保持しておき、前記要素の再送が要求された場合は、前記リトライバッファに保持した要素を前記メインメモリ制御部へ再送してもよく、

各非同期動作ユニットで再送を開始する要素のうち、最も若い要素番号に基づいて再送を開始する要素の要素番号を修正してもよい。

【 0 0 2 7 】

さらに、前記VSCアドレスバッファのオーバーフローと前記非同期動作ユニットからの要素の再送を繰り返し実行するデッドロック状態を検出した場合に、前記非同期動作ユニットからの要素の再送タイミングをずらすための遅延値を前記メインメモリ制御部から前記非同期動作ユニットへ送出し、

前記非同期動作ユニットは、前記要素の再送タイミングを前記メインメモリ制

御部から受信した該遅延値だけ遅らせてもよい。

【0028】

上記のような情報処理装置及びそのメモリ制御方法では、ベクトル・スキヤッタ命令により非同期動作ユニットからそれぞれ発行される複数の要素の書き込みリクエストのうち、格納先アドレスが一致し、かつ要素番号順に処理されていない場合に、要素番号が若い要素の書き込みリクエストに対応して生成される書き込み許可信号の出力を抑止することで、複数の非同期動作ユニット間で発生する処理順序の逆転による影響を排除することが可能になる。

【0029】

また、VSCアドレスバッファで複数の非同期動作ユニットから送信される格納先アドレスをそれぞれ保持し、VSCアドレスバッファがオーバーフローを起こした場合に、該オーバーフローさせたベクトル・スキヤッタ命令を発行した非同期動作ユニットへ要素の再送を要求するVSCアドレスバッファ制御部を備え、非同期動作ユニットは、自ユニットが発行した前記ベクトル・スキヤッタ命令で指定された各要素をそれぞれ保持するリトライバッファを有し、メインメモリ制御部から要素の再送が要求された場合にリトライバッファに保持した要素をメインメモリ制御部へ再送することで、VSCアドレスバッファを効率よく使用することができる。

【0030】

【発明の実施の形態】

次に本発明について図面を参照して説明する。

【0031】

(第1の実施の形態)

図1は本発明の情報処理装置の第1の実施の形態の構成を示すブロック図であり、図2は図1に示したMMCの一構成例を示すブロック図である。

【0032】

図1に示すように、本実施形態の情報処理装置は、CPU100と主記憶装置であるMMU (Main Memory Unit) 200とを有する構成である。

【0033】

CPU100は、Master-Unit1とSlave-Unit2の2つの非同期動作ユニットを備えている。Master-Unit1及びSlave-Unit2は、Master-Unit1でベクトル命令の発行／終了を管理することを除けば、基本的にベクトル命令にしたがって同じ処理を実行するユニットである。これら2つのベクトルユニットは、通常、異なる2つのLSIで構成されるが、非同期で動作するユニットであれば、同一のLSI内に形成されていてもよい。なお、図1では、CPU100が1つのMaster-Unit1と1つのSlave-Unit2とで構成される例を示しているが、CPU100は、1つのMaster-Unit1と複数のSlave-Unit2で構成されていてもよく、1つのMaster-Unit1と少なくとも1つのSlave-Unit2から成る組を複数備えた構成であってもよい。

【0034】

Master-Unit1は、命令制御部11、Master側RQ制御回路12、ベクトルレジスタ13、及びPNU14を有し、Slave-Unit2は、Slave側RQ制御回路22、ベクトルレジスタ23、及びPNU24を有する構成である。

【0035】

図1に示したMaster-Unit1及びSlave-Unit2は、VSC命令により256個の要素を一度に処理する構成例であり、Master-Unit1は偶数番号の要素を処理し、Slave-Unit2は奇数番号の要素を処理するものとする。

【0036】

命令制御部11は、ベクトル命令の発行時に、Master側RQ制御回路12及びSlave側RQ制御回路22にそれぞれ対応するオペコードを供給する。

【0037】

Master側RQ制御回路12は、VSC命令のオペコードを受け取ると、VSC命令の実行条件を全てチェックし、処理を開始するためのVSCstart信号を所定のタイミングでSlave側RQ制御回路22へ送出する。また、偶数番号の各要素の要素リクエストを発行し、ベクトルレジスタ13から読み出した、VSC命令で指定されたメモリへの格納先アドレス及びデータと共に、要素番号、ユニット番号CPU#、及び識別子VSCid等を含むリクエスト情報をPNU14へ送出する。なお、ユニット番号CPU#は、要素リクエストの発行元ユニッ

トを区別するためのものであり、例えば、Master-Unit 1 から送出されるリクエスト情報のユニット番号 CPU # には偶数番号が割り当てられ、Slave-Unit 2 から送出されるリクエスト情報のユニット番号 CPU # には奇数番号が割り当てられる。但し、CPU が複数組の Master-Unit 1 と Slave-Unit 2 で構成される場合は、一組の Master-Unit 1 と Slave-Unit 2 に割り当てるユニット番号 CPU # は連続する値に設定する。また、識別子 V S C i d は、V S C 命令を区別するためのものであり、1 つの V S C 命令で発行される要素リクエストには全て同一の識別子が付与される。

【 0 0 3 8 】

P N U 1 4 は、Master 側リクエスト制御回路 1 2 から受け取った要素リクエスト、リクエスト情報、格納先アドレス及びデータを CPU 出力 R Q レジスタ 1 5 で一旦保持した後、M M U 2 0 0 へ送出する。

【 0 0 3 9 】

Slave 側 R Q 制御回路 2 2 は、上記 Master 側 R Q 制御回路 1 2 と同様に奇数番号の各要素の処理を実行すると共に、一つの V S C 命令で発行される複数の要素リクエストのうち、最終の要素リクエストの発行タイミングで処理の終了を示す V S C e n d 信号を Master 側 R Q 制御回路 1 2 へ送信する。V S C e n d 信号を受け取った Master 側 R Q 制御回路 1 2 は、自身で発行する最終の要素リクエストの発行を待って命令制御部 1 1 から発行された次の V S C 命令の処理を開始する。

【 0 0 4 0 】

Slave-Unit 2 が有するベクトルレジスタ 2 3 は、Master-Unit 1 が有するベクトルレジスタ 1 3 と同様に動作し、Slave-Unit 2 が有する P N U 2 4 は、Master-Unit 1 が有する P N U 1 4 と同様に動作する。

【 0 0 4 1 】

一方、M M U 2 0 0 は、データを格納する M M (Main Memory) 4 と M M 4 に対するデータの書き込み動作を制御する M M C (Main Memory Controller) 3 とを有する構成である。なお、図 1 では、M M U 2 0 0 を 1 つだけ備えた構成を示しているが、M M U 2 0 0 の数は 1 つに限定されるものではなく、複数の M M U

2 0 0 を有していてもよい。

【 0 0 4 2 】

MM4 は、並列にアクセス可能なバンクと呼ばれる複数（図 1 では 8 個）のメモリユニットから構成され、各バンクには指定されたアドレスにしたがってインターリーブでデータが格納される。

【 0 0 4 3 】

MMC 3 は、MMU 入力 RQ レジスタ 3 1₁、3 1₂、VSC アドレスバッファ 3 2、リクエスト分配回路 3 3、順序監視回路 3 4、WE レジスタ 3 5、及び a d r / d a t a レジスタ 3 6 を備えた構成である。

【 0 0 4 4 】

MMU 入力 RQ レジスタ 3 1₁、3 1₂ は、Master-Unit 1 及び Slave-Unit 2 から送出されたリクエスト情報、格納先アドレス及びデータを受信するレジスタであり、Master-Unit 1 及び Slave-Unit 2 に対応して 2 つ設けられている。

【 0 0 4 5 】

VSC アドレスバッファ 3 2 は、MM4 に対する各要素の格納先アドレスを一時的に保持するレジスタであり、図 1 に示した Master-Unit 1 及び Slave-Unit 2 はそれぞれ 2 5 6 個の要素を一度に処理するため、それに対応して 2 5 6 W（ワード）のアドレスを保持するレジスタを 2 セット備えている。本実施形態の VSC アドレスバッファ 3 2 は、1 つの VSC 命令に対応する格納先アドレスのみ保持し、複数の VSC 命令に対応する格納先アドレスをオーバーラップして格納することはない。

【 0 0 4 6 】

リクエスト分配回路 3 3 は、要素の格納先アドレスに対応する MM4 のバンクに対して、書き込みを許可する M e m - W r i t e 信号、並びに指定された格納先アドレス及びデータをそれぞれ分配する。

【 0 0 4 7 】

順序監視回路 3 4 は、1 つの VSC 命令で発行される要素リクエストのうち、格納先アドレスが一致し、かつ要素番号順に処理されていない要素リクエストがある場合に、要素番号が若い方の要素リクエストで生成される M e m - W r i t

e 信号の出力を抑止する。

【 0 0 4 8 】

WE レジスタ 3 5 は、リクエスト分配回路 3 3 から送出された Mem - W r i t e 信号をそれぞれ保持し、格納先アドレスに対応する MM 4 のバンクへ送出する。

【 0 0 4 9 】

a d r / d a t a レジスタ 3 6 は、リクエスト分配回路 3 3 から送出された格納先アドレス及びデータをそれぞれ保持し、格納先アドレスで指定された MM 4 のバンクへ送出する。

【 0 0 5 0 】

図 2 に示すように、順序監視回路 3 4 は、先行する要素リクエストと後続の要素リクエストの要素番号及び格納先アドレスをそれぞれ比較する比較回路 3 7 と、比較回路 3 7 の比較結果にしたがって、処理の順序が逆転し、かつ格納先アドレスが一致している場合に、要素番号が若い要素リクエストで生成される Mem - W r i t e 信号のリクエスト分配回路 3 3 への出力を抑止する WE 抑止回路 3 8 と、V S C アドレスバッファ 3 2 に対する格納先アドレスの書き込み処理を制御するバッファ制御回路 3 9 とを有する構成である。

【 0 0 5 1 】

比較回路 3 7 は、V S C アドレスバッファ 3 2 に格納された先行する要素リクエストと後続の要素リクエストの要素番号を比較し、処理順序の逆転を検出する要素番号比較器 3 7 1 と、V S C アドレスバッファ 3 2 に格納された先行する要素リクエストと後続の要素リクエストの格納先アドレスを比較し、同一のアドレスに対するアクセスを検出するアドレス比較器 3 7 2 と、要素番号比較器 3 7 1 及びアドレス比較器 3 7 2 の比較結果の論理積を出力する論理積ゲート 3 7 3 とを備えている。要素番号比較器 3 7 1、アドレス比較器 3 7 2、及び論理積ゲート 3 7 3 は、V S C アドレスバッファ 3 2 に格納可能なワード数（エントリ数）に対応して、それぞれ 256×2 個 = 512 個ずつ備えている。

【 0 0 5 2 】

なお、アドレス比較器 3 7 1 は、後続の要素リクエストが Master-Unit 1 から

発行されている場合はSlave-Unit 2 から発行された先行する要素リクエストとアドレスどうしを比較する。また、後続の要素リクエストがSlave-Unit 2 から発行されている場合はMaster-Unit 1 から発行された先行する要素リクエストとアドレスどうしを比較する。これは、Master-Unit 1 またはSlave-Unit 2 から連続して発行される要素リクエストでは処理の順序が逆転しないことによる。

【0053】

論理積ゲート 373 の出力がアクティブになったとき、Master-Unit 1 及びSlave-Unit 2 から発行された要素リクエストの処理の順序が逆転し、MM4 に対する不正な書き込み処理が発生することを示している。

【0054】

バッファ制御回路 39 は、書き込み制御部 391、デコード回路 392、及びVSCidレジスタ 393、VSCid比較器 394、及びインバータ 395 を備えている。

【0055】

デコード回路 392 は、Master-Unit 1 及びSlave-Unit 2 から送信されたオペコードをデコードし、VSC命令であることを示すVSC信号、及びMM4 に対するデータの書き込み許可信号であるMem-Write信号を生成する。

【0056】

書き込み制御部 (Write cont) 391 は、要素が有効であることを示すバリッド (V) 信号、ユニット番号CPU#、及びVSC信号を入力とし、要素が有効であるとき、要素リクエストの発行元に対応するVSCアドレスバッファ 32 内をサーチし、書き込み可能な (空いている) アドレスとライトイネーブルWE信号とを該VSCアドレスバッファ 32 へ送出する論理回路である。

【0057】

VSCアドレスバッファ 32 は、書き込み制御部 391 から送出されたアドレス及びWE信号にしたがって、MMU入力RQレジスタ 31₁、31₂から転送された格納先アドレス (Address) を要素番号 (ELM) に関連付けてそれぞれ格納する。

【0058】

VSCidレジスタ393は、先行する要素リクエストに付与された識別子VSCidを保持し、VSCid比較器394は、VSCidレジスタ393で保持された識別子VSCidと後続の要素リクエストに付与された識別子VSCidとを比較する。そして、後続の識別子VSCidが変化したとき（次のVSC命令の処理に移行したとき）、VSCアドレスバッファ32に格納された内容をクリアするためのタイミング信号（Clear）をインバータ395を介して送出する。

【0059】

WE抑止回路38は、VSCアドレスバッファ32の数に対応して、論理和ゲート381、第1の論理積ゲート382、及び第2の論理積ゲート383をそれぞれ2個ずつ備えている。

【0060】

論理和ゲート381は、比較回路37の各論理積ゲート373から出力される比較結果から、要素が有効であり、同一のVSC命令であり、かつ識別子VSCidが変化しない条件下で、対応する256個の論理積ゲート373のいずれかの出力がアクティブになった場合に、MM4に対するデータの書き込み許可信号であるMem-Write信号の出力を抑止する。

【0061】

リクエスト分配回路33は、Master-Unit1及びSlave-Unit2に対応して生成された2つのMem-Write信号をMM4の対応するバンクへそれぞれ送出するためのスイッチであるWE用2×8スイッチ331と、Master-Unit1及びSlave-Unit2から送信された格納先アドレス及びデータをMM4の対応するバンクへそれぞれ送出するためのスイッチであるadr/data用2×8スイッチ332とを備えている。

【0062】

次に、本実施形態の情報処理装置のVSC命令発行時の動作について図面を用いて説明する。

【0063】

図3は図1に示した情報処理装置のVSC命令による動作の様子を示す模式図

である。なお、図 3 は CPU 1 0 0 が 1 つの Master-Unit 1 と 1 つの Slave-Unit 2 とで構成される場合の動作を示している。

【 0 0 6 4 】

図 3 に示すように、Master-Unit 1 の命令制御部 1 1 から V S C 命令が発行されると、Master側 R Q 制御回路 1 2 は V S C s t a r t 信号を生成し、Slave側 R Q 制御回路 2 2 へ該 V S C s t a r t 信号を送出する。

【 0 0 6 5 】

続いて、Master側 R Q 制御回路 1 2 及び Slave側 R Q 制御回路 2 2 は、自ユニットが担当する要素番号の要素リクエストをそれぞれ発行し、ベクトルレジスタ 1 3、2 3 から V S C 命令で指定された格納先アドレス及びデータを読み出し (REG read)、読み出した格納先アドレス及びデータと生成したリクエスト情報とを P N U 1 4、2 4 を介して M M U 2 0 0 へ送付する。

【 0 0 6 6 】

ここでは、Master-Unit 1 が要素 0 から順に要素 n まで偶数番号の要素リクエストを発行し、Slave-Unit 2 が要素 1 から順に要素 n - 1 まで奇数番号の要素リクエストを発行するものとする。

【 0 0 6 7 】

例えば、要素 n - 1 と要素 n の要素リクエストにおいて、同一のアドレスにデータを格納する指示があった場合、M M U 2 0 0 では要素 n の要素リクエストを先行して受信する可能性がある。

【 0 0 6 8 】

本実施形態では、後続の要素 n - 1 の要素リクエストに対応する要素番号及び格納先アドレスと、先行する要素 n の要素リクエストに対応する要素番号及び格納先アドレスとを M M C 3 の比較回路 3 7 でそれぞれ比較する。そして、要素 n の格納先アドレスが V S C アドレスバッファ 3 2 に先に格納されたことを検出した場合は、要素 n - 1 の要素リクエストで生成される M e m - W r i t e 信号の送出を W E 抑止回路 3 8 により抑止する。したがって、要素 n - 1 のデータが M M 4 へ書き込まれないため、要素 n のデータが要素 n - 1 のデータで上書きされることがない。

【 0 0 6 9 】

一方、Slave-Unit 2 で最終の要素リクエストが発行されると、Slave側 R Q 制御回路 2 2 は、その発行タイミングで Master側 R Q 制御回路 1 2 へ処理の終了を示す V S C e n d 信号を送信する。V S C e n d 信号を受信した Master側 R Q 制御回路 1 2 は、自身の最終要素リクエストの発行終了を待って命令制御部 1 1 から発行された次の V S C 命令の処理を開始する。

【 0 0 7 0 】

MMC 2 0 0 は、次の V S C 命令による要素リクエストを受信すると、先行する V S C 命令で格納された V S C アドレスバッファ 3 2 の内容を全てクリアする。このとき、先行する V S C 命令と後続する V S C 命令の区別は識別子である V S C i d が用いられる。

【 0 0 7 1 】

なお、本実施形態では、説明を容易にするために C P U が 1 組の Master-Unit 1 と Slave-Unit 2 で構成される例を示したが、実際の情報処理装置では C P U が複数組の Master-Unit 1 と Slave-Unit 2 を備えている場合が多い。

【 0 0 7 2 】

例えば、C P U を複数組の Master-Unit 1 と Slave-Unit 2 で構成し、さらに V S C アドレスバッファ 3 2 で複数の V S C 命令に対応する格納先アドレスを保持する構成を採用すると、V S C 命令の発行間隔を短縮できるため情報処理装置の処理速度をさらに向上させることができる。

【 0 0 7 3 】

しかしながら、そのような構成では、V S C アドレスバッファ 3 2 が (Master-Unit と Slave-Unit の組数) × 2 個だけ必要になり、そのエントリ数は、Master-Unit 及び Slave-Unit がそれぞれ 2 5 6 個の要素を一度に処理すると仮定すると、以下の式で示す数だけ必要になる。

【 0 0 7 4 】

エントリ数 = (Master-Slave の組数) × 5 1 2 × (V S C 命令のオーバーラップ数)

また、Master-Unit 1 と Slave-Unit 2 の組数を増やすと、比較回路 3 7 の各構

成要素もそれに伴って追加しなければならないため回路量が大幅に増加する。特にVSCアドレスバッファで複数のVSC命令に対応する格納先アドレスを保持する構成では、処理が複雑になることによる回路増加も加わるため、さらなる回路量の増加を招いてしまう。

【0075】

したがって、VSCアドレスバッファ32は、1つのVSC命令分の格納先アドレスを保持する構成が好ましい。また、Master-Unit及びSlave-Unitの組数は、VSC命令の発行間隔が広がることによる性能低下と回路量の増加とをトレードオフすることで、情報処理装置に要求される性能を満たす最適な組数に設定することが望ましい。

【0076】

同様に、CPUが1つのMaster-Unit1と複数のSlave-Unit2とを有する構成でも、VSCアドレスバッファ32は1つのVSC命令分の格納先アドレスを保持する構成が好ましく、Slave-Unit2の数は情報処理装置に要求される性能を満たす最適な値に設定することが望ましい。

【0077】

本実施形態の情報処理装置によれば、複数のユニットから発行される要素リクエストに対する処理順序の逆転による影響を排除できるため、CPUを複数の非同期動作ユニットで構成しても不正な書き込み処理が発生することが無い。したがって、CPUを複数のユニットで構成することで、各ユニットの回路規模を小さくすることが可能になり、ユニット(LSI)の歩留まりや外部端子数が低減されて、コストが低減される。

【0078】

(第2の実施の形態)

次に本発明の情報処理装置の第2の実施の形態について図面を用いて説明する。

【0079】

図4は本発明の情報処理装置の第2の実施の形態のCPU構成を示すブロック図であり、図5は本発明の情報処理装置の第2の実施の形態のMMC構成を示す

ブロック図である。また、図6は図5に示したデッドロック検出制御部の構成を示すブロック図であり、図7は図6に示したデッドロック検出制御部の動作の様子を示す模式図である。

【0080】

第1の実施の形態では、Master-Unit及びSlave-Unitに対応してVSCアドレスバッファを1つずつ設けているため、MMCはMaster-Unit及びSlave-Unitの数だけVSCアドレスバッファを備える必要がある。第2の実施の形態のMMCは、Master-Unit及びSlave-Unitの要素リクエストで指定される格納先アドレスを1つのVSCアドレスバッファで保持する構成である。また、それに対応してMaster-Unit及びSlave-Unitから送出されるリクエスト情報、格納先アドレス及びデータを、1つのMMU入力RQレジスタで共通に受信する。さらに、本実施形態のVSCアドレスバッファは、複数のVSC命令に対応する格納先アドレスをそれぞれ保持する構成である。

【0081】

このような構成では、格納先アドレスの保持に必要なVSCアドレスバッファの空きエントリが不足する（オーバーフロー）可能性があるため、オーバーフローさせたVSC命令に対応する要素を再送させるためのRQ再送要求信号をMMCからMaster-Unit及びSlave-Unitへそれぞれ送信する。

【0082】

一方、第2の実施の形態のMaster-Unit及びSlave-Unitは、既に発行した要素リクエストに対応するリクエスト情報、格納先アドレス及びデータを保持するためのリトライバッファを備え、MMCからRQ再送要求信号を受信すると、該リトライバッファで保持したリクエスト情報、格納先アドレス及びデータをMMCへ再送する。その際、再送を開始する要素番号がMaster-Unit及びSlave-Unitでずれている可能性があるため、Master-Unit及びSlave-Unitは、再送を開始する要素番号のうち、要素番号が若い方の要素から再送を開始する。例えば、Master-Unitの要素番号の方が若い場合、Slave-UnitはMaster-Unitの要素番号+1の要素番号から再送を開始する。また、Slave-Unitの要素番号の方が若い場合、Master-UnitはSlave-Unitの要素番号-1の要素番号から再送を開始する。

【0083】

さらに、本実施形態のMaster-Unit及びSlave-Unitは、MMCからRQ再送要求信号を受信すると、VSCアドレスバッファをオーバーフローさせたVSC命令の要素リクエストに対応するVSCアドレスバッファの内容をクリアするためのバッファクリア信号をMMCへ送出する。

【0084】

図4に示すように、本実施形態のMaster-Unit5は、発行した要素リクエストに対応するリクエスト情報を保持するための制御用リトライバッファ56と、該要素リクエストに対応する格納先アドレス及びデータを保持するためのアドレス・データ用リトライバッファ57と、MMCからRQ再送要求信号を受信した場合に、リクエスト情報、格納先アドレス及びデータをMMCへ再送するための第1のセクタ58及び第2のセクタ59とを有する構成である。

【0085】

Master側RQ制御回路52は、発行した要素リクエストに対応するリクエスト情報を制御用リトライバッファ56及び第1のセクタ58へそれぞれ送出する。第1のセクタ58は、RQ再送要求信号を受信していない場合はMaster側RQ制御回路52から受信したリクエスト情報をPNU54へ送出し、RQ再送要求信号を受信した場合は制御用リトライバッファ56で保持したリクエスト情報をPNU54へ送出する。同様に、ベクトルレジスタ53は、要素リクエストで指定された格納先アドレス及びデータをアドレス・データ用リトライバッファ57及び第2のセクタ59へそれぞれ送出する。第2のセクタ59は、RQ再送要求信号を受信していない場合はベクトルレジスタ53から受け取った格納先アドレス及びデータをPNU54へ送出し、RQ再送要求信号を受信した場合はアドレス・データ用リトライバッファ57で保持した格納先アドレス及びデータをPNU54へ送出する。

【0086】

また、Master側RQ制御回路52は、MMCからRQ再送要求信号を受信すると、Slave-Unit6のSlave側RQ制御回路62に対して、再送を開始する要素の要素番号であるstart要素番号を取得するためのstart要素番号指示を送信する。Slave

e側RQ制御回路62は、Master側RQ制御回路52からstart要素番号指示を受信すると、Slave-Unit6で再送を開始する要素のstart要素番号をMaster側RQ制御回路52へ返送する（start要素番号報告）。Master側RQ制御回路52は、Master-Unit5で再送を開始するstart要素番号とSlave側RQ制御回路62から受信したstart要素番号とを比較し、若い方の要素番号に基づいて再送を開始する要素番号を修正する。また、必要に応じてSlave側RQ制御回路62へ再送を開始する要素番号の修正結果を送信する。そして、自ユニットで発行した、VSCアドレスバッファをオーバーフローさせたVSC命令の要素リクエストに対応するVSCアドレスバッファの内容をクリアするためのバッファクリア信号をPNU54を介してMMCへ送出する。なお、PNU54は、リクエスト情報、格納先アドレス及びデータを再送する前にバッファクリア信号をMMCへ送出する。

【0087】

Slave-Unit6は、上記Master側RQ制御回路52及びSlave側RQ制御回路62の動作の違いを除けばMaster-Unit5と同様の構成である。

【0088】

図5に示すように、本実施形態のMMC7が有するMMU入力RQレジスタ71は、図2に示した第1の実施の形態のMMCが有するMMU入力RQレジスタに、CPUから送信されるバッファクリア信号を受信するためのV2フィールドを追加した構成である。

【0089】

また、本実施形態のVSCアドレスバッファ72は、ユニット番号CPU#、及びVSC命令を識別するための識別子VSCidを、要素番号（ELM）及び格納先アドレス（Address）に関連付けてそれぞれ保持する構成である。なお、本実施形態では、後述するデッドロック状態を回避するためにVSCアドレスバッファ72のエントリ数を（Master-Unit5で一度に処理するワード数）+1（本実施形態では257ワード）に設定する。

【0090】

さらに、本実施形態のMMC7は、図2に示した第1の実施の形態のMMCに、VSCアドレスバッファ制御部80、再送リクエストレジスタ81、及びデッ

ドロック検出制御部 82 を追加した構成である。

【0091】

VSC アドレスバッファ制御部 80 は、VSC アドレスバッファ 72 のオーバーフローを検出したときに、対応するリクエスト情報、格納先アドレス及びデータを Master-Unit 5 及び Slave-Unit 6 から再送させるための再送要求信号を再送リクエストレジスタ 81 及びデッドロック検出制御部 82 へそれぞれ送出する。

【0092】

デッドロック検出制御部 82 は、VSC アドレスバッファ 72 のオーバーフローと Master-Unit 5 及び Slave-Unit 6 からの要素の再送を繰り返し実行するデッドロック状態を検出し、デッドロック状態が発生した場合に要素の再送タイミングをずらすための遅延値 (BSYCNT) を再送リクエストレジスタ 81 へ送出する。

【0093】

再送リクエストレジスタ 81 は、MMU 入力 RQ レジスタ 71 からユニット番号 CPU #、識別子 VSC id、及び要素番号 ELM が要素リクエストの発行タイミングに同期して転送され、VSC アドレスバッファ制御部 80 から再送要求信号を受信すると、VSC アドレスバッファ 72 をオーバーフローさせた要素リクエストの発行元の Master-Unit 5 及び Slave-Unit 6 に RQ 再送要求信号を送出する。その際、デッドロック検出制御部 82 から受信した遅延値 BSYCNT を RQ 再送要求信号に添付して送出する。

【0094】

本実施形態の比較回路 77 は、第 1 の実施の形態で示した要素番号比較器、及びアドレス比較器に加えて、VSC アドレスバッファ 72 に格納された先行する要素リクエストと後続の要素リクエストに対応するユニット番号 CPU # どうしを比較するユニット番号比較器 771 と、VSC アドレスバッファ 72 に格納された先行する要素リクエストと後続の要素リクエストに対応する識別子 VSC id どうしを比較する識別子比較器 772 とを備えている。

【0095】

ユニット番号比較器 771 及び識別子比較器 772 の比較結果は第 1 の論理積ゲート 773 を介してバッファ制御回路 79 が有する第 2 の論理積ゲート 791

へ出力される。第2の論理積ゲート791は、MMU入力RQレジスタ71のV2フィールドから出力されるバッファクリア信号と第1の論理積ゲート773の出力信号との論理積を出力する。

【0096】

バッファクリア信号をMaster-Unit5またはSlave-Unit6から受信した場合、バッファ制御回路79は、先行する要素リクエストと後続の要素リクエストに対応するユニット番号CPU#及び識別子VSCidに一致するVSCアドレスバッファ72のエントリをクリアする。その他の構成及び動作は第1の実施の形態と同様であるため、その説明は省略する。なお、図5に示すMMC7は、リクエスト分配回路、WEレジスタ、及びadr/dataレジスタを省略した構成を示している。

【0097】

次に、上述したデッドロック状態について説明する。

【0098】

例えば、CPUが1つのMaster-Unit5と1つのSlave-Unit6とを備え、Master-Unit5及びSlave-Unit6がそれぞれ256個の要素を一度に処理する構成の場合、信号の伝搬遅延や要素リクエストの様々な発行制限により、Master-Unit5がSlave-Unit6よりも先行して256個の要素リクエストを発行する可能性がある。さらに、これらの要素リクエストが全て同じMMC7に対して発行された場合、VSCアドレスバッファ72のエントリ数を256とすると、Master-Unit5から発行された要素リクエストの処理だけでVSCアドレスバッファ72の全てのエントリが使用される。この状態では、Slave-Unit6から発行される最初の要素リクエストの処理でVSCアドレスバッファ72がオーバーフローするため、Master-Unit5及びSlave-Unit6にRQ再送要求信号が送信され、最初の要素から再送が開始される。このとき、再度、Master-Unit5がSlave-Unit6よりも先行して256個の要素リクエストを発行すると、VSCアドレスバッファ72のオーバーフローとMaster-Unit5及びSlave-Unit6の再送処理とが繰り返し実行されるため、VSC命令の処理が永久に進まない状態に陥る。この状態をデッドロックと言う。

【0099】

このようなデッドロック状態を回避するためには、VSCアドレスバッファ72のエントリ数を、(Master-Unit5で一度に処理する要素数) + 1 (本実施形態では257) だけ確保すればよい。

【0100】

一方、CPUが複数組のMaster-Unit5及びSlave-Unit6を備え、Master-Unit5及びSlave-Unit6でそれぞれ256個の要素を一度に処理する構成の場合も、上記と同様に各組毎にMaster-Unit5がSlave-Unit6よりも先行して256個の要素リクエストを発行する可能性がある。そのため、(Master-Unit5で一度に処理する要素数) × (Master-Unit5とSlave-Unit6の組数) よりもVSCアドレスバッファ72のエントリ数が少なければ、VSCアドレスバッファ72のオーバーフローと要素の再送処理とを繰り返す(以下、再送サイクルと称す) デッドロック状態に陥る可能性がある。

【0101】

この場合、デッドロック状態を回避するためには、VSCアドレスバッファ72のエントリ数を (Master-Unit5で一度に処理する要素数) × (Master-Unit5とSlave-Unit6の組数) + 1 だけ確保すればよい。

【0102】

しかしながら、実際には殆ど発生しないデッドロック状態を回避するためだけにVSCアドレスバッファ72のエントリ数を増加させることは、ハードウェアの増加を招くため、MMUのコストが増大する新たな問題が発生する。

【0103】

本実施形態では、ハードウェアの増加を抑制しつつデッドロック状態を回避するために、図5に示したデッドロック検出制御部82をMMC7に備えている。

【0104】

デッドロック検出制御部82は、所定の期間内に、同じMaster-Unit5及びSlave-Unit6から、同じVSC命令に対応する同じ要素番号が再送された場合に、2度目の再送時にMaster-Unit5及びSlave-Unit6による要素リクエストの発行タイミングを遅らせ、要素リクエストの再送サイクルを乱すことでデッドロック

から脱出させるためのものである。

【0105】

図6に示すように、デッドロック検出制御部82は、デッドロック状態であるか否かを検出するためのデッドロック検出期間を計数するカウンタ回路821と、再送された格納先アドレス及びデータに対応するユニット番号CPU#、識別子VSCid、及び要素番号ELMについて、それぞれの直前の値と比較し、それらが全て一致する場合にデッドロック状態と判定するデッドロック判定回路822と、再送時にMaster-Unit5及びSlave-Unit6による要素リクエストの発行タイミングをずらすための遅延値BSYCNTを生成する乱数発生回路823と、カウンタ回路821及びデッドロック判定回路822の動作を制御する検出期間制御回路824とを有する構成である。

【0106】

カウンタ回路821は、セレクタ、レジスタ（検出期間cnt）、減算器（-1）、及び比較器（=0）を備え、カウント開始時に外部から供給されるデッドロック検出期間信号の値（定数）を検出期間cntに取り込み、その値からカウント（decrement）を開始し、カウント値=0を検出したときにその結果を検出期間制御回路824へ出力する。

【0107】

デッドロック判定回路822は、再送idレジスタ（REG）及び比較器（=）を備え、要素の再送時にMMU入力RQレジスタ71から転送されるCPU#／VSCid／ELMを再送idREGで保持し、次の再送時に転送されるCPU#／VSCid／ELMと再送idREGに保持した値とを比較し、それらが全て一致した場合にデッドロック検出信号をアクティブにする。

【0108】

検出期間制御回路824は、ラッチ回路、論理和ゲート、及び論理積ゲートを備え、デッドロック判定回路822によりCPU#／VSCid／ELMの一致が検出された場合にカウンタ回路821にカウントを開始させ、カウント終了時（デッドロック検出期間の終了時）、あるいはデッドロック検出信号のアクティブ時でカウンタ回路821の検出期間cnt、及びデッドロック判定回路の再送

i d R E G をリセットする。

【0109】

このような構成では、図7に示すように、任意の組のMaster-Unit5及びSlave-Unit6から発行された要素リクエストによりVSCアドレスバッファ72がオーバーフローし、要素の再送が発生（RQ再送発生）すると、検出期間制御回路824がデッドロック検出中信号をアクティブに設定し、カウンタ回路821にカウントを開始させてデッドロックの検出処理を開始する。デッドロックの検出処理はカウンタ回路821の検出期間c n t の出力値が0になったときに終了する。

【0110】

デッドロック検出期間長は、1組のMaster-Unit5及びSlave-Unit6で発生する再送サイクルの周期の最長値（定数）に設定され、その値を、例えば情報処理装置が備える不図示のROMに格納しておく。そして、情報処理装置の電源投入時にデッドロック検出期間信号として該ROMからカウンタ回路821へ供給する。

【0111】

デッドロック判定回路822は、デッドロック検出処理の開始時に再送された要素リクエストに対応するユニット番号CPU#、識別子VSC i d、及び要素番号ELMを再送i d R E Gに保存しておく。そして、2回目の再送時のユニット番号CPU#、識別子VSC i d、及び要素番号ELMと再送i d R E Gに保持した値とを比較し、それらの値が全て一致したときにデッドロック状態であると判定する（デッドロック検出）。

【0112】

このとき、デッドロック判定回路822の出力信号であるデッドロック検出信号がアクティブになり、乱数発生回路823で生成された遅延値BSYCNTが再送リクエストレジスタ81に格納される。再送リクエストレジスタ81は、RQ再送要求信号に該遅延値BSYCNTを添付し、Master-Unit5及びSlave-Unit6に送出する（RQ再送要求）。同時に、検出期間制御回路824はデッドロック検出中信号をリセットして処理を終了する。

【0113】

RQ再送要求信号を受信したMaster-Unit 5のMaster側RQ制御回路52及びSlave-Unit 6のSlave側RQ制御回路62は、遅延値BSYCNTだけ要素リクエストの発行タイミングをずらし、指定された要素を再送する。

【0114】

このような処理を行うことで、1回目の再送サイクルよりも2回目の再送サイクルが長くなり再送サイクルの繰り返し周期が乱れるため、デッドロック状態から抜け出ることができる。

【0115】

但し、上述したデッドロックの回避方法は、複数組のMaster-Unit 5及びSlave-Unit 6間で発生する再送サイクルの周期を乱す方法であるため、CPUが1つのMaster-Unitと1つのSlave-Unitで構成される場合に発生するデッドロック状態は回避することができない。したがって、VSCアドレスバッファ72のエントリ数は、少なくとも(Master-Unit 5で一度に処理する要素数) + 1だけ必要である。

【0116】

本実施形態の情報処理装置によれば、VSCアドレスバッファ72を効率よく使用することが可能になり、VSCアドレスバッファ72の数を低減できるため、MMU 7のコストを低減することができる。

【0117】

【発明の効果】

本発明は以上説明したように構成されているので、以下に記載する効果を奏する。

【0118】

ベクトル・スキッタ命令により非同期動作ユニットからそれぞれ発行される複数の要素の書き込みリクエストのうち、格納先アドレスが一致し、かつ要素番号順に処理されていない場合に、要素番号が若い要素の書き込みリクエストに対応して生成される書き込み許可信号の出力を抑止することで、複数の非同期動作ユニット間で発生する処理順序の逆転による影響を排除することが可能になる。

したがって、CPUを複数の非同期動作ユニットで構成できるため、各ユニットの規模を小さくすることが可能になり、ユニット（LSI）の歩留まりや外部端子数が低減されて、コストが低減される。

【0119】

また、VSCアドレスバッファで複数の非同期動作ユニットから送信される格納先アドレスをそれぞれ保持し、VSCアドレスバッファがオーバーフローを起こした場合に、該オーバーフローさせたベクトル・スキヤッタ命令を発行した非同期動作ユニットへ要素の再送を要求するVSCアドレスバッファ制御部を備え、非同期動作ユニットは、自ユニットが発行した前記ベクトル・スキヤッタ命令で指定された各要素をそれぞれ保持するリトライバッファを有し、メインメモリ制御部から要素の再送が要求された場合にリトライバッファに保持した要素をメインメモリ制御部へ再送することで、VSCアドレスバッファを効率よく使用することができるため、VSCアドレスバッファの低減が可能になり、MMUのコストを低減することができる。

【図面の簡単な説明】

【図1】

本発明の情報処理装置の第1の実施の形態の構成を示すブロック図である。

【図2】

図1に示したMMCの一構成例を示すブロック図である。

【図3】

図1に示した情報処理装置のVSC命令による動作の様子を示す模式図である。

【図4】

本発明の情報処理装置の第2の実施の形態のCPU構成を示すブロック図である。

【図5】

本発明の情報処理装置の第2の実施の形態のMMC構成を示すブロック図である。

【図6】

図5に示したデッドロック検出制御部の構成を示すブロック図である。

【図7】

図6に示したデッドロック検出制御部の動作の様子を示す模式図である。

【図8】

ベクトル方式の計算機で使用されるVSC命令の仕様を示す模式図である。

【図9】

CPUがMaster-Unit及びSlave-Unitから構成される場合にVSC命令により要素が主記憶装置へ格納される様子を示す模式図である。

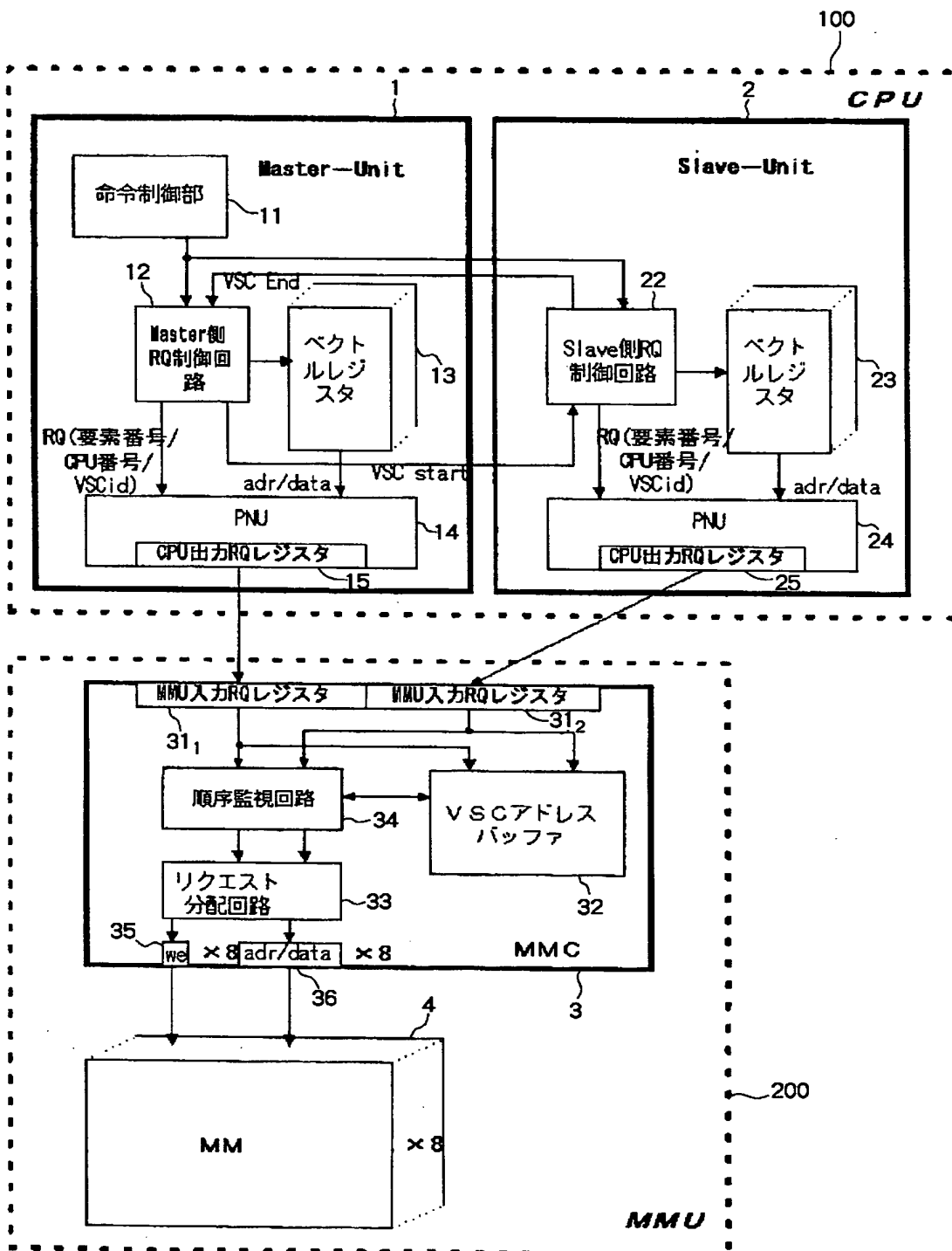
【符号の説明】

- 1、 5 Master-Unit
- 2、 6 Slave-Unit
- 3、 7 MMC
- 4 MM
- 1 1 命令制御部
- 1 2、 5 2 Master側RQ制御回路
- 1 3、 2 3、 5 3 ベクトルレジスタ
- 1 4、 2 4、 5 4 PNU
- 1 5、 2 5 CPU出力RQレジスタ
- 2 2 Slave側リクエスト制御回路
- 3 1₁、 3 1₂、 7 1 MMU入力RQレジスタ
- 3 2、 7 2 VSCアドレスバッファ
- 3 3 リクエスト分配回路
- 3 4 順序監視回路
- 3 5 WEレジスタ
- 3 6 adr/dataレジスタ
- 3 7、 7 7 比較回路
- 3 8、 7 8 WE抑止回路
- 3 9、 7 9 バッファ制御回路
- 5 4 PNU

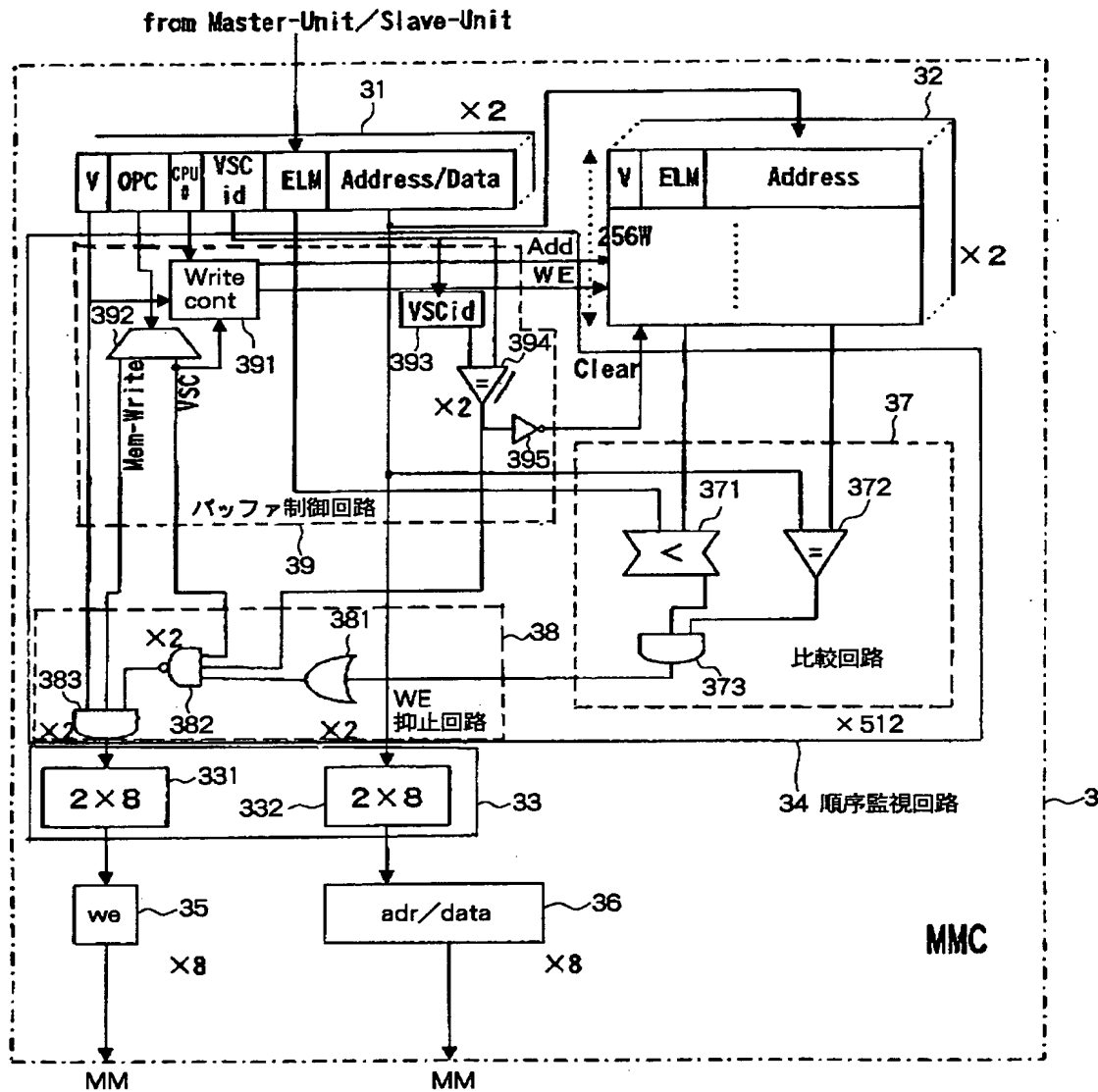
56	制御用リトライバッファ
57	アドレス・データ用リトライバッファ
58	第1のセレクタ
59	第2のセレクタ
80	VSCアドレスバッファ制御部
81	再送リクエストレジスタ
82	デッドロック検出制御部
100	CPU
200	MMU
371	要素番号比較器
372	アドレス比較器
373	論理積ゲート
381	論理和ゲート
382、773	第1の論理積ゲート
383、791	第2の論理積ゲート
391	書き込み制御部
392	デコード回路
393	VSCidレジスタ
394	VSCid比較器
395	インバータ
331	WE2×8スイッチ
332	adr/data2×8スイッチ
771	ユニット番号比較器
772	識別子比較器
821	カウンタ回路
822	デッドロック判定回路
823	乱数発生回路
824	検出期間制御回路

【書類名】 図面

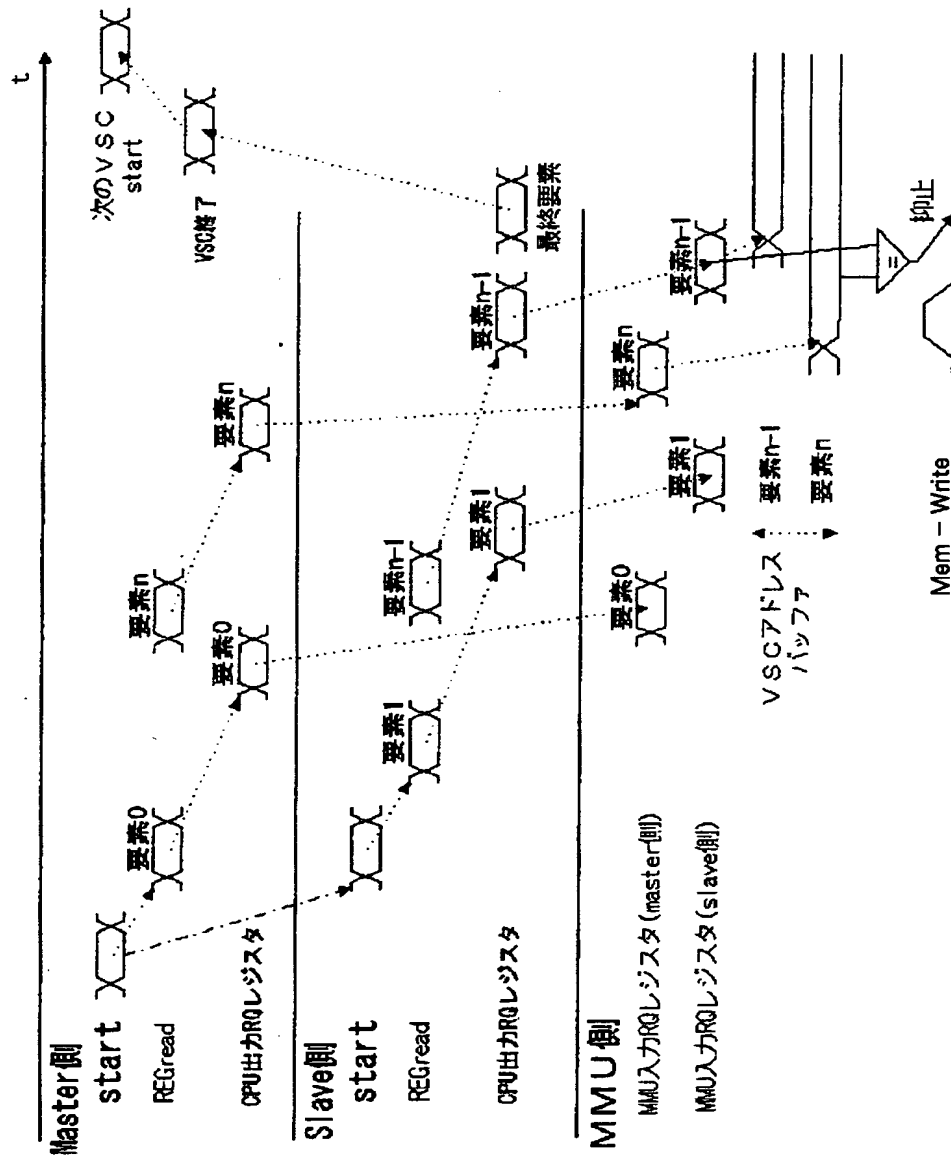
【図1】



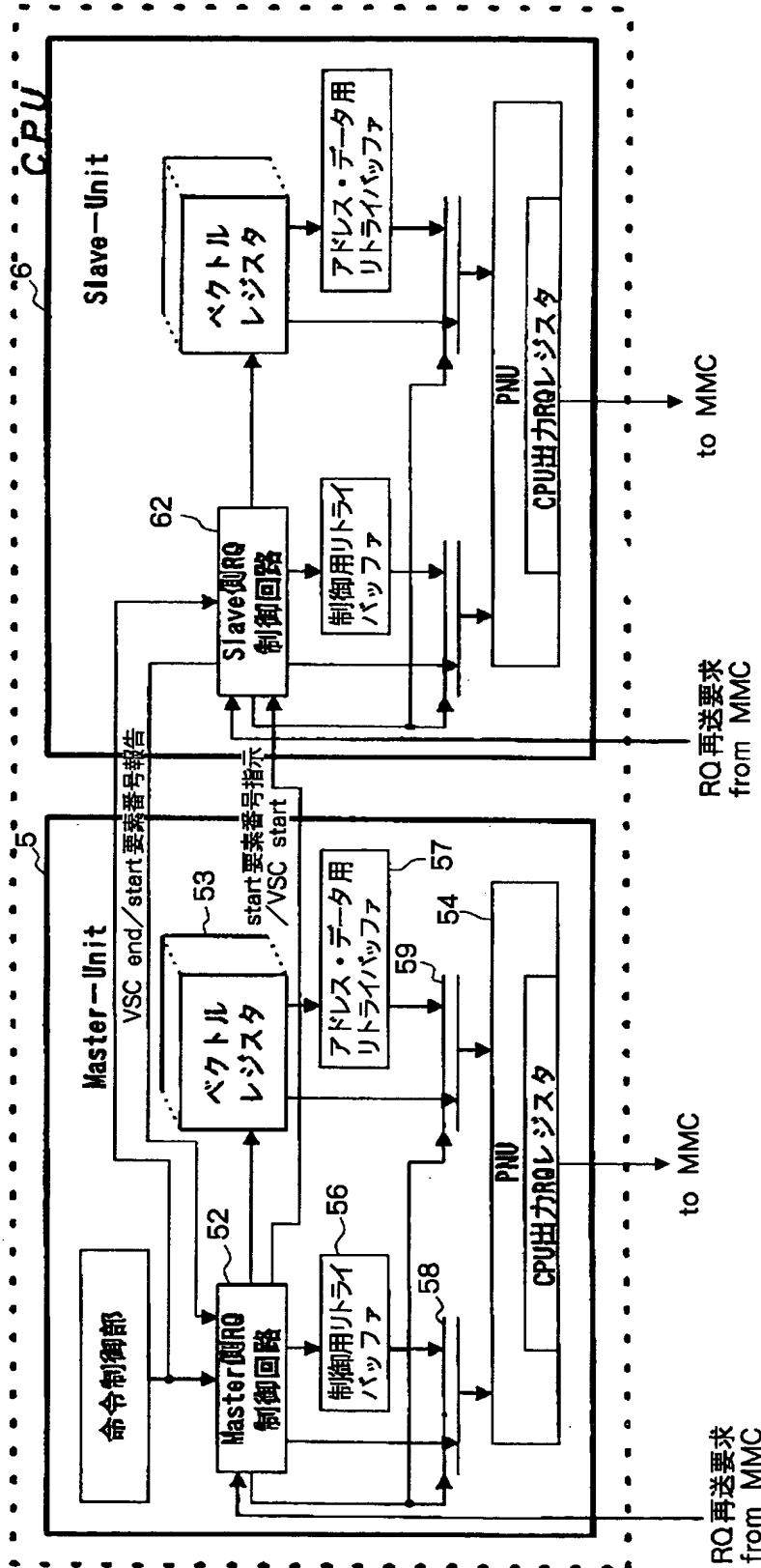
【図2】



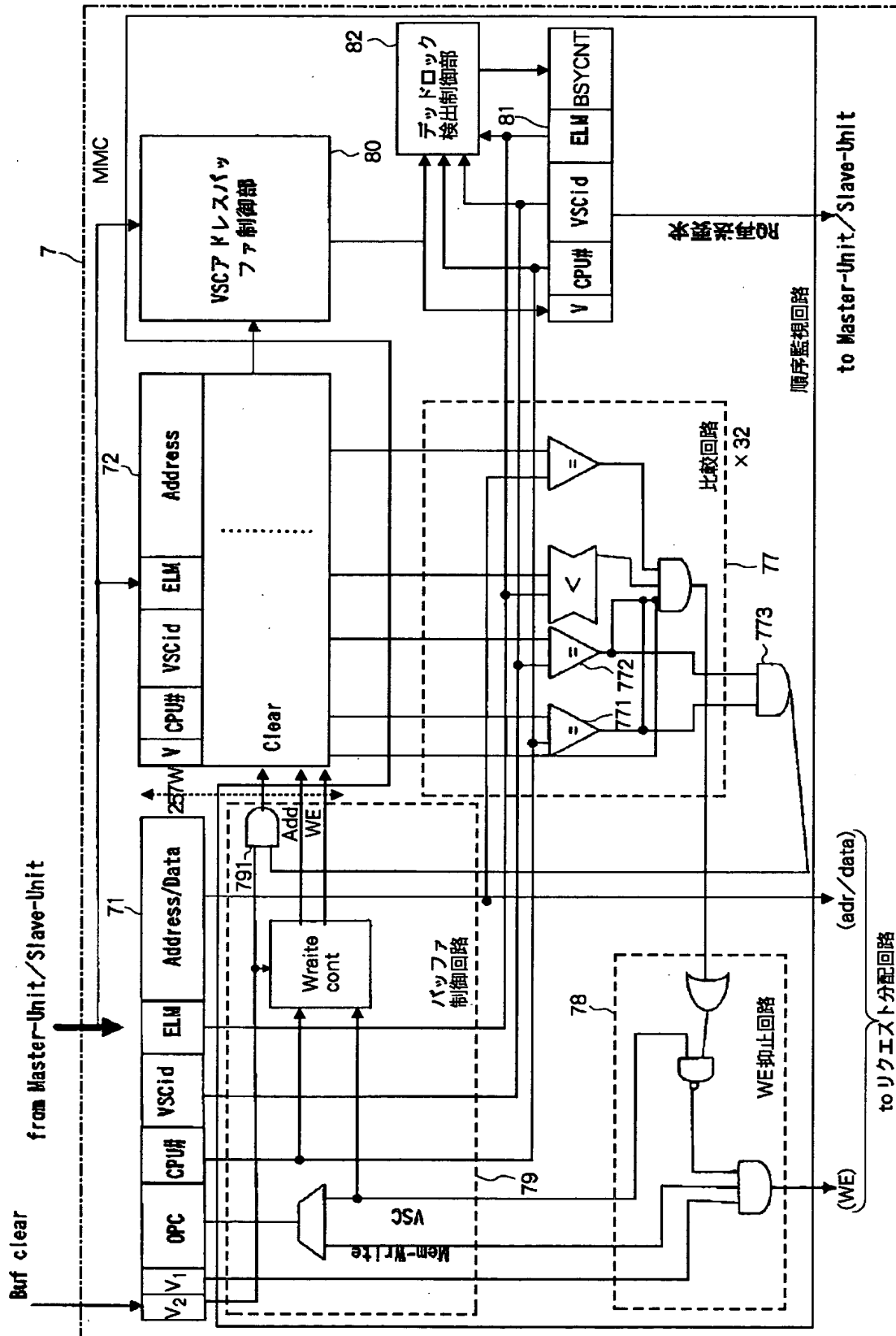
【図 3】



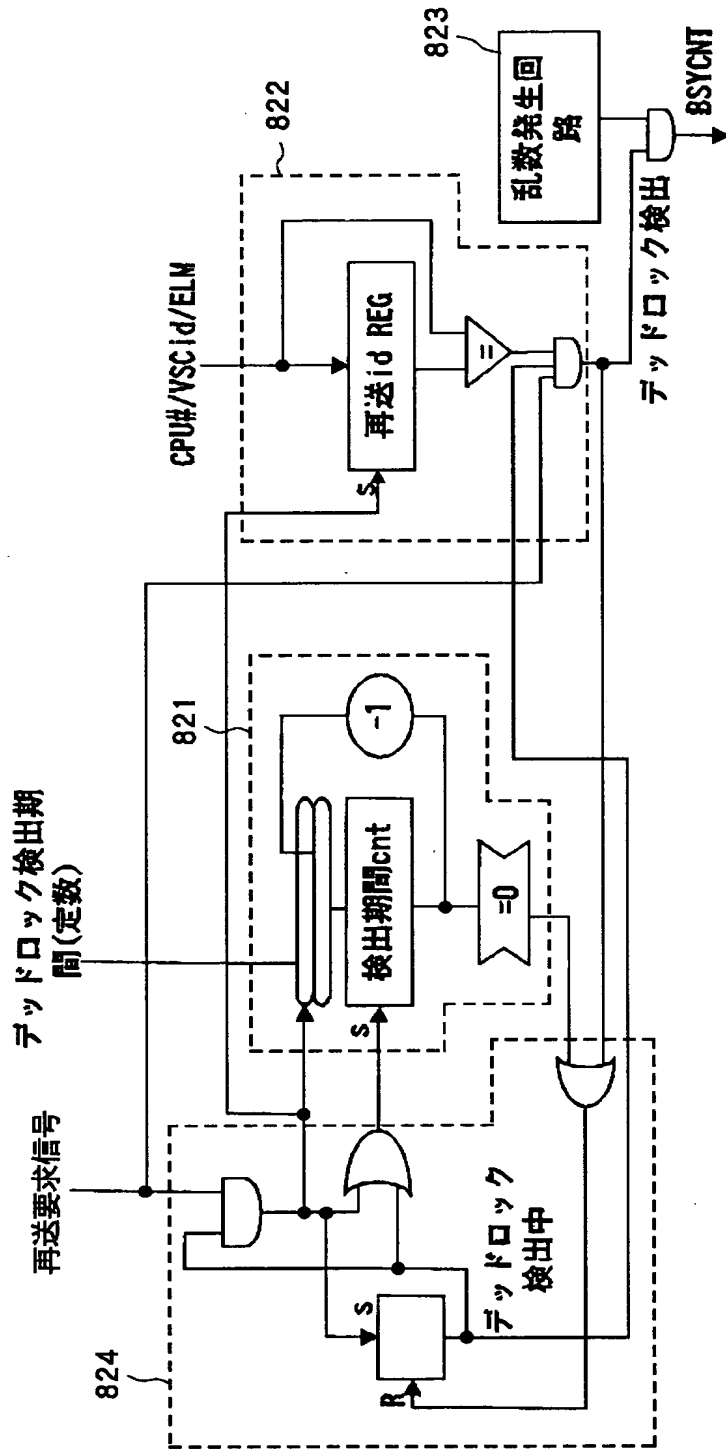
【図4】



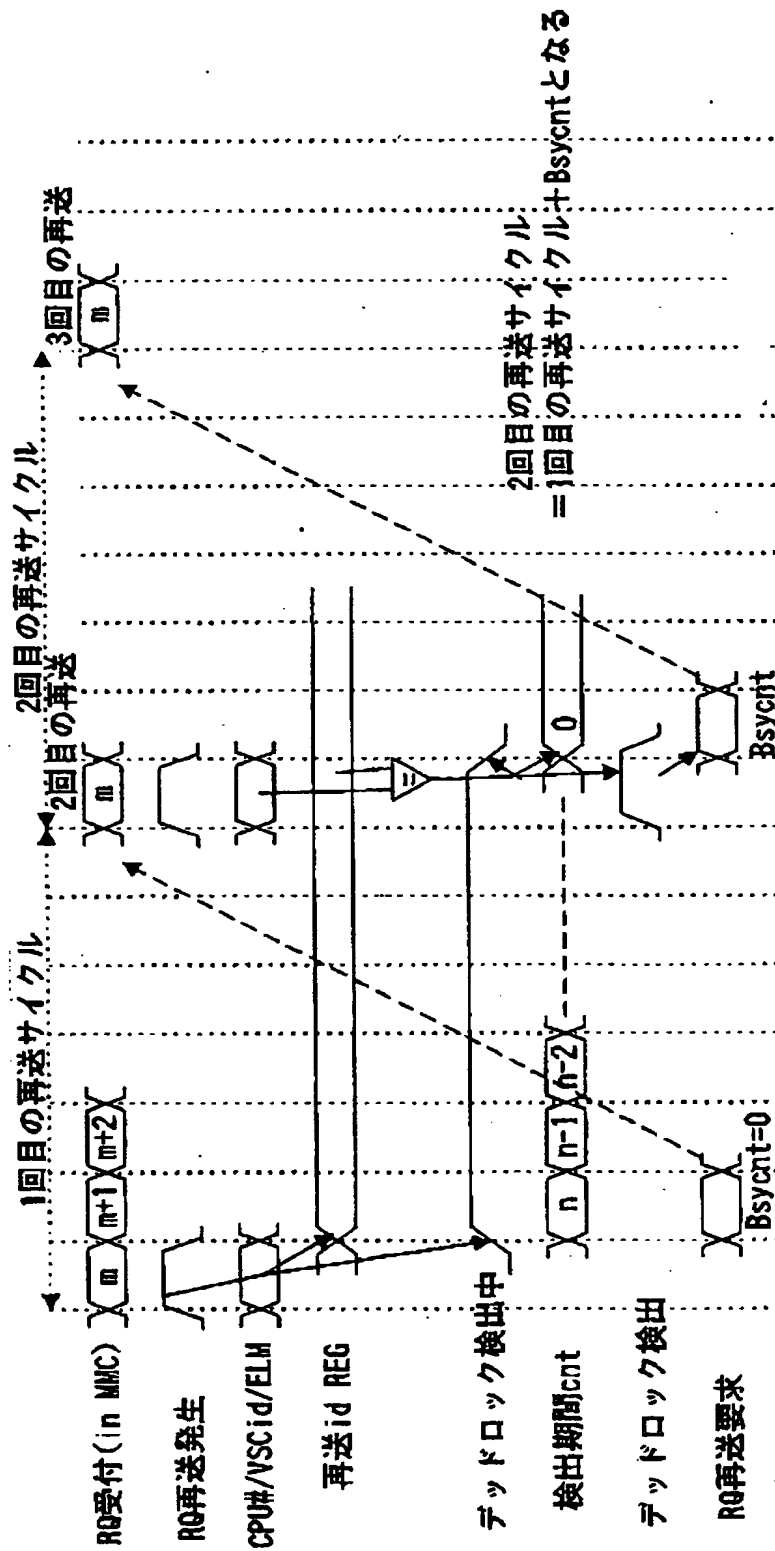
【図5】



【図 6】



【図 7】



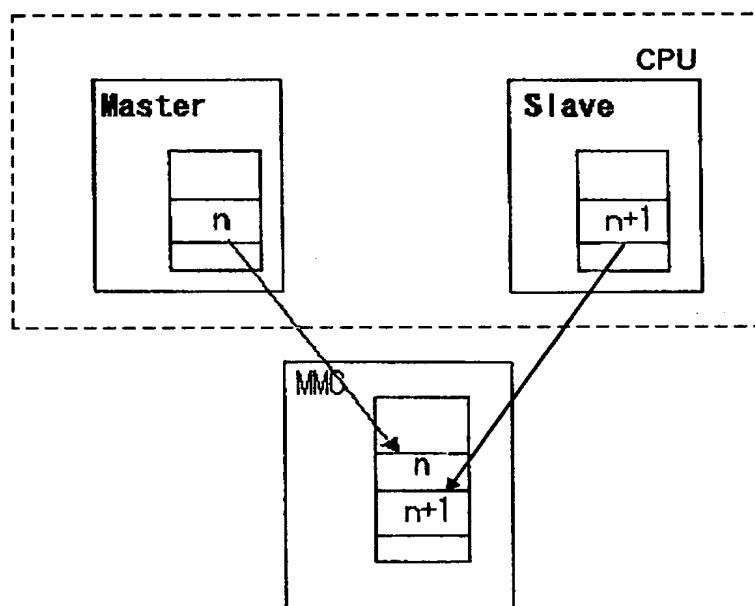
【図 8】

VSC命令仕様

OPC	x	y	z
-----	---	---	---

$M(V_y(0), V_y(1), \dots, V_y(VL-1)) \leftarrow V_z$

【図 9】



【書類名】 要約書

【要約】

【課題】 CPUが複数の非同期動作ユニットで構成される場合に、各ユニット間の処理順序の逆転による影響を排除できる情報処理装置及びそのメモリ制御方法を提供する。

【解決手段】 複数の非同期動作ユニットから構成されるCPUと、データを格納するメインメモリ部と、ベクトル・スキヤッタ命令で指定された各要素のメインメモリ部への格納先アドレスを保持するVSCアドレスバッファ、及びベクトル・スキヤッタ命令で非同期動作ユニットからそれぞれ発行されるメインメモリ部に対する要素の書き込みリクエストのうち、格納先アドレスが一致し、かつ要素番号順に処理されていない場合に、要素番号が若い要素の書き込みリクエストに対応して生成されるメインメモリ部への書き込み許可信号の出力を抑止する順序監視回路を備えたメインメモリ制御部とを有する構成とする。

【選択図】 図1

出 願 人 履 歴 情 報

識別番号 [000004237]

1. 変更年月日 1990年 8月29日

[変更理由] 新規登録

住 所 東京都港区芝五丁目7番1号

氏 名 日本電気株式会社